

AMENDMENT TO THE CLAIMS

Claims pending

- At time of the Action: Claims 1-32.
- After this Response: Claims 1-4, 6, 10-20, 22-23, 25-28, 30, and 32.

Canceled or Withdrawn claims: 5, 7-9, 21, 24, 29, and 31

Amended claims: 1, 6, 10, 14, 17, 23, 25, 28, 30, and 32

New claims: None

1. (Currently Amended) A method comprising:

representing hardware and software resources of a distributed computer system as model components, wherein the model components are selected from a group comprising:

a module that is representative of a behavior of the application that is

implemented using the hardware and software resources;

a port that is representative of a service access point for the module;

and

a wire that is representative of an allowable communication

connection between two or more ports; and

forming, from the model components, a logical scale-independent model of an application to be implemented by the distributed computer system.

2. (Original) A method as recited in claim 1, wherein each model component represents one or more similar resources.

1 3. **(Original)** A method as recited in claim 1, wherein each model
2 component is depicted in a graphical user interface as a graphical icon.

3
4 4. **(Original)** A method as recited in claim 1, wherein the model
5 components have an associated schema that specifies the hardware and software
6 resources represented by the model components.

7
8 5. **(Canceled)**

9
10 6. **(Currently Amended)** A method as recited in claim 1, wherein the
11 ~~model components comprise~~ the group of the model components further comprises
12 a store that is representative of persistent data storage.

13
14 7. **(Canceled)**

15
16 8. **(Canceled)**

17
18 9. **(Canceled)**

19
20 10. **(Currently Amended)** A method as recited in claim [[9]] 1, wherein
21 the group of the model components further comprises at least one of:

22 a store that is representative of persistent data storage;

23 an event source that is representative of a logical connection point for the
24 module or the store from which event messages originate;

1 an event sink that is representative of a logical connection point for the
2 module or the store to receive the event messages; and

3 an event wire that is representative of an interconnection between the event
4 source and the event sink.

5
6 11. (Original) A method as recited in claim 1, further comprising
7 storing the scale-independent model on a computer-readable medium.

8
9 12. (Original) A method as recited in claim 1, further comprising
10 converting the scale-independent model into a blueprint of the server data center,
11 the blueprint specifying the hardware and software resources used to physically
12 implement the application.

13
14 13. (Original) A computer-readable medium storing computer-
15 executable instructions that, when executed on a computer, perform the method of
16 claim 1.

17
18 14. (Currently Amended) A method comprising:
19 defining individual model components as abstract functional operations that
20 are physically implemented by one or more computers and one or more software
21 programs executing on the computers, the model components having an associated
22 schema dictating how the functional operations are specified;
23 interconnecting the model components to logically connect the functional
24 operations; and
25

1 generating a scale-independent application from the interconnected model
2 components and the associated schema; and
3 converting the scale-independent application into a blueprint that specifies
4 the computers and the software programs used to physically implement the
5 application.

6
7 15. (Original) A method as recited in claim 14, wherein each model
8 component is depicted in a graphical user interface as a graphical icon.

9
10 16. (Original) A method as recited in claim 14, wherein the defining
11 comprises entering, via a user interface, a description of elements needed to
12 implement the functional operations.

13
14 17. (Currently Amended) A method as recited in claim 14, wherein
15 each ~~graphical resource~~ model component represents a set of resources provided
16 by the computers and the software programs, the resources being scalable from
17 one to many.

18
19 18. (Original) A method as recited in claim 14, wherein the model
20 components are selected from a group comprising:

21 a module that is representative of a behavior of the application;

22 a port that is representative of a communication access point for the
23 module; and

24 a wire that is representative of an interconnection between two or more
25 ports.

1
2 19. (Original) A method as recited in claim 18, wherein the group of the
3 model components further comprises:

4 a store that is representative of persistent data storage;

5 an event source that is representative of a logical connection point for the
6 module or the store from which event messages originate;

7 an event sink that is representative of a logical connection point for the
8 module or the store to receive the event messages; and

9 an event wire that is representative of an interconnection between the event
10 source and the event sink.
11

12 20. (Original) A method as recited in claim 14, further comprising
13 storing the application on a computer-readable medium.
14

15 21. (Canceled)
16

17 22. (Original) A computer-readable medium storing computer-
18 executable instructions that, when executed on a computer, perform the method of
19 claim 14.
20

21 23. (Currently Amended) A method comprising:
22 representing hardware and software resources of a distributed computer
23 system as model components, wherein the model components are selected from a
24 group comprising:
25

1 a module that is representative of a behavior that is implemented
2 using the hardware and software resources;

3 a port that is representative of a communication access point for the
4 module; and

5 a wire that is representative of an interconnection between two or
6 more ports; and

7 associating the model components with a schema dictating how the
8 hardware and software resources are specified.

9
10 24. (Canceled)

11
12 25. (Currently Amended) A method as recited in claim [[24]] 23,
13 wherein the group of the model components further comprises:

14 a store that is representative of persistent data storage;

15 an event source that is representative of a logical connection point for the
16 module or the store from which event messages originate;

17 an event sink that is representative of a logical connection point for the
18 module or the store to receive the event messages; and

19 an event wire that is representative of an interconnection between the event
20 source and the event sink.

21
22 26. (Original) A method as recited in claim 23, further comprising
23 creating a scale-independent application from the model components and the
24 associated schema.
25

1 27. (Original) A method as recited in claim 26, further comprising
2 converting the scale-independent application into a blueprint that specifies the
3 hardware and software resources used to physically implement the application on
4 the distributed computer system.

5
6 28. (Currently Amended) A modeling system, comprising:
7 a set of model components that represent hardware and software resources
8 of a distributed computer system;
9 a schema associated with the model components that dictate how the
10 resources are specified; and
11 a user interface to enable a developer to create an application by selecting
12 and interconnecting the model components and specifying the functionality of the
13 model components in accordance with the schema; and
14 a converter to convert the application to a blueprint that specifies the
15 hardware and software resources used to physically implement the application on
16 the distributed computer system.

17
18 29. (Canceled)

19
20 30. (Currently Amended) A computer-readable medium comprising
21 computer-executable instructions that, when executed on one or more processors,
22 direct a computing device to:
23 represent hardware and software resources of a distributed computer system
24 as model components;

1 associate the model components with a schema dictating how the hardware
2 and software resources are specified; and

3 create an application by specifying the functionality of the model
4 components in accordance with the schema and interconnecting the model
5 components; and

6 convert the application to a blueprint that specifies the hardware and
7 software resources used to physically implement the application on the distributed
8 computer system.

9
10 31. (Canceled)

11
12 32. (Currently Amended) A system comprising:

13 means for representing hardware and software resources of a distributed
14 computer system;

15 means for specifying how the resources represented by the model
16 components are specified; and

17 means for selecting and interconnecting the model components and
18 specifying the functionality of the model components to create an application; and

19 means for converting the application to a blueprint that specifies the
20 hardware and software resources used to physically implement the application on
21 the distributed computer system.